

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
Before the Board of Patent Appeals and Interferences

In re Patent Application of

Atty Dkt. SCS-550-619

C# M#

Confirmation No. 4576

EVRARD et al.

TC/A.U.: 2183

Serial No. 10/527,812

Examiner: K. Vicary

Filed: June 14, 2005

Date: May 19, 2008

Title: PROCESSING ACTIVITY MASKING IN A DATA PROCESSING SYSTEM



Mail Stop Appeal Brief - Patents

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

Sir:

☐ **Correspondence Address Indication Form Attached.**

☐ **NOTICE OF APPEAL**

Applicant hereby **appeals** to the Board of Patent Appeals and Interferences
from the last decision of the Examiner twice/finally rejecting
applicant's claim(s).

\$510.00 (1401)/\$255.00 (2401) \$

☐ An appeal **BRIEF** is attached in the pending appeal of the
above-identified application

\$510.00 (1402)/\$255.00 (2402) \$

☐ Credit for fees paid in prior appeal without decision on merits

-\$ ()

☒ A reply brief is attached.

(no fee)

☐ Petition is hereby made to extend the current due date so as to cover the filing date of this
paper and attachment(s)

One Month Extension \$120.00 (1251)/\$60.00 (2251)

Two Month Extensions \$460.00 (1252)/\$230.00 (2252)

Three Month Extensions \$1050.00 (1253)/\$525.00 (2253)

Four Month Extensions \$1640.00 (1254)/\$820.00 (2254) \$

☐ "Small entity" statement attached.

Less month extension previously paid on

-\$ ()

TOTAL FEE ENCLOSED \$ 0.00

☐ **CREDIT CARD PAYMENT FORM ATTACHED.**

Any future submission requiring an extension of time is hereby stated to include a petition for such time extension.
The Commissioner is hereby authorized to charge any deficiency, or credit any overpayment, in the fee(s) filed, or
asserted to be filed, or which should have been filed herewith (or with any paper hereafter filed in this application by this
firm) to our **Account No. 14-1140**. A duplicate copy of this sheet is attached.

901 North Glebe Road, 11th Floor
Arlington, Virginia 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100
SCS:kmm

NIXON & VANDERHYE P.C.

By Atty: Stanley C. Spooner, Reg. No. 27,393

Signature: _____



**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
Before the Board of Patent Appeals and Interferences**

In re Patent Application of

EVARD et al.

Atty. Ref.: SCS-550-619

Serial No. 10/527,812

TC/A.U.: 2183

Filed: June 14, 2005

Examiner: K. Vicary

For: **PROCESSING ACTIVITY MASKING IN A DATA PROCESSING
SYSTEM**

* * * * *

May 19, 2008

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

REPLY BRIEF

This Reply Brief is responsive to the Examiner's Answer mailed March 18, 2008 and the various new points of argument raised therein. Appellants will respond to each of the Examiner's new points of argument beginning with section 11 on page 8 in the order in which it is presented.

**Appeal Brief Section A. General differences between the
claimed invention and the Qiu reference (U.S. Patent 6,844,782)**

Appellants' Appeal Brief points out the differences between two different methods of analysis, one of which, the Qiu reference can defend against and both of which the claimed invention can defend against.

The Examiner on page 9, first full paragraph of the Examiner's Answer, basically attempts to lump the two different methods of attack, i.e., simple power analysis (SPA) and differential power analysis (DPA) into the same category of "power analysis." However, as explained in Appellants' specification Background of the Invention, encryption algorithms such as disclosed in the Qiu patent are effective to defeat SPA but are still susceptible to DPA analysis determining when data is written to a particular register and when it is not.

Therefore, the Examiner appears not to recognize the difference between the Qiu system in which a simple algorithm is sufficient to defend against simple power analysis (SPA) and is still susceptible to differential power analysis (DPA) and the present invention which cannot be defeated by either simple power analysis or differential power analysis.

In lumping the two very different types of analysis into the general category of "power analysis," the Examiner simply indicates that he does not appreciate the difference in analysis between SPA and DPA. Therefore he cannot possibly appreciate that while Qiu is effective to prevent SPA, it is still susceptible to DPA which, after all, is the problem to be solved by the present invention as discussed in Appellants' originally filed application (page 1, lines 7-17 and elsewhere).

The Examiner also somewhat disingenuously states that "the independent claim describes an invention in which a conditional operation writes a result to a trash register instead of a data processing register when a conditional write data processing operation is resolved to not normally write." The Examiner apparently ignores the fact that

Appellants' independent claim lays out the function of a "conditional-write data processing instruction" as "encoding condition codes specifying conditions under which said conditional-write data processing instruction will or will not be permitted to write data to effect a change in state of said processor core."

The difference between this and the Qiu reference is important, since the distinction is made between the defenses against SPA (wherein encoding/decoding algorithm is crafted to try to disguise the results of hidden processing) and defenses against DPA (where intrinsic weaknesses of an encoding/decoding algorithm are addressed, and even a custom algorithm cannot circumvent).

Accordingly, the Examiner's defining a more general "power analysis" rather than Appellants' specifically disclosed simple power analysis (SPA) and differential power analysis (DPA) blurs the distinctions between the two. He fails to appreciate that Appellants' specification as originally filed admits that the problem in the prior art defenses is that they are capable of defending against SPA but not DPA and the present invention serves to defend against both SPA and DPA.

By analogy, the Examiner's argument is that radio signal receivers are known as evidenced by Qui's teaching of an AM receiver and that this somehow renders obvious all other radio receivers such as FM receivers. Qui's teaching of how to defeat simple power analysis does not render obvious the claimed solution to defeat differential power analysis even though both Qui and the present invention defeat power analysis in general.

On page 10, section 13 of the Examiner's Answer, the Examiner also attempts to respond to the points made in Appeal Brief section A by dismissing the assertion that Qiu

only addresses SPA whereas the present invention addresses DPA. The Examiner correctly notes that the present claims and specification do not explicitly make use of the terms SPA and DPA. These terms are well known in the art to identify the two different manners of power analysis – simple power analysis and differential power analysis. It should also be noted that whether or not these terms are used in Appellants' specification or claims has no bearing on whether the present invention as specified by the combination of structures in the claims provides a system which will defeat DPA.

Additionally, the Examiner's observation, while literally correct, does not affect the fact that one of ordinary skill in the art would immediately recognize the Qiu reference as an algorithm based system which will only defeat SPA, but can be analyzed by using DPA systems. Further, there is no requirement for the terms SPA or DPA to be included in the claims and those of ordinary skill in the art will understand and appreciate that the structures and the structural interconnections set out in the claims will not only effectively defeat SPA, but also defeat DPA, which Qiu clearly will not.

The burden on the Examiner is not to find references that mention SPA or DPA or power analysis in general, or to analyze claims which may or may not include these terms, but instead to determine how or where the Qiu reference anticipates or discloses the recited claim elements/method steps and claimed interrelationships between elements which are specified in Appellants' independent claims. As noted in the Appeal Brief, the Examiner has simply failed to meet this burden of establishing where these structures exist in the Qiu reference.

Appeal Brief Section B. The Examiner fails to identify any structure in the Qiu reference which comprises Appellants' claimed "trash register"

The Examiner misunderstands the well-known terms of "instruction" and "condition code." The Examiner's misuse of these terms with respect to the well-known definitions extends throughout the "Response to Argument" portion of the Examiner's Answer.

In section 13, the Examiner asserts that Qiu discloses "condition codes" encoded in "an instruction" which determine whether or not a state changing write is permitted or not. Firstly, the determining factor in Qiu is the value of a particular bit of a private key and not a "condition code" as the term is known by those of ordinary skill in the art. The conventional definition of an instruction is that it is a "single operation of a processor." (see attached Wikipedia definition of "Instruction"). There is no reason one of ordinary skill in the art would equate an instruction with Qui's multiple lines of high level program code.

Additionally, as noted above, Qiu does not disclose a "conditional-write data processing instruction" and instead teaches a section of high-level program code which calls a further routine. Because the Examiner does not distinguish between systems capable of defeating SPA and DPA (as opposed to Qiu which can only defeat SPA), the Qiu reference does not need the use of a "trash register" in order to defeat the simpler form of power analysis, i.e., SPA. Thus, the fact that Qiu does not teach the claimed "trash register" structure recited in Appellants' independent claims is of no concern to the

Examiner. The Examiner has simply erred in drafting power analysis so broadly as to encompass any power analysis and then he assumes that if Qiu defeats the simplest power analysis (SPA), by definition it must somehow disclose the claimed structures or combinations of structures which are designed to defeat the more complex differential power analysis (DPA). Of course, there is no evidence of record to support these suppositions and conclusion.

Finally, regarding paragraph 13, the Examiner alleges that there “may be aspects of the instant specification which overcome differential power analysis,” but again the Examiner is simply incorrect. It is clear that the Qiu reference relies upon a programmer to implement the SPA defense by coding the algorithm that is disclosed. As this is the well-known manner of defeating simple power analysis (SPA), this is well known in the art.

However, the defense disclosed in the present invention and recited in claim 1 is independent of, and indeed hidden from, any programmer. By the claims’ encapsulating of the necessary function within an instruction, the presently claimed invention does not rely on the programmer for implementation and thus addresses weaknesses intrinsic to algorithms themselves. The claimed invention of independent claims 1 and 6 is clearly not merely an SPA defense, but falls within the realm of a DPA defense, i.e., one which can mask operation whether or not data is written to the processor.

In sections 14 and 15 of the Examiner’s Answer, the Examiner contends that Qiu at column 6, lines 9-19, discloses the use of a “trash register.” Even assuming for the purpose of argument that there is a disclosure of what Appellants have claimed to be a

“trash register,” there is no indication that the Examiner provides any of the interrelationships specified in Appellants’ independent claims, i.e., to which result data values are written instead of a data processing register “upon execution of said conditional-write data processing instruction when said condition codes within said conditional-write data processing instruction do not permit a write to effect a change in state of said processor core.”

Appeal Brief Sections C & D.

The Examiner appears to focus on one portion of Appellants’ argument at the end of the last full paragraph on page 11 of the Appeal Brief, i.e., “Qiu simply fails to disclose any concept of a single instruction being executed in one of two different ways.” While the well-known definition of “instruction” clearly makes this a correct statement, the Examiner has apparently been distracted from the key point in section D of the Appeal Brief, i.e., the Examiner’s failure to appreciate that Qiu teaches away from the present invention by requiring an unnecessary mathematical operation. Appellants’ claim does not teach the unnecessary mathematical operation noted in the Qiu reference because the invention is encapsulated within an instruction, as noted above. Because the function is encapsulated within an instruction, there is no need for programmer implementation.

Even if one assumes for the purpose of argument that Qui does teach a trash register, there is no suggestion that Qui teaches other structures and interrelationships which are specified in the independent claims.

In the last paragraph of section 16 of the Examiner's Answer, the Examiner states that he is "unsure as to how Appellant's invention would prevent differential power analysis." While "prevent" is not the correct word because simple power analysis or differential power analysis could be applied to any systems, the present invention does defeat DPA because the invention is not dependent upon implementation by a programmer and instead provides an instruction that itself ensures that detail of its execution is not detectable, even by differential power analysis (DPA). The Examiner's suggestion that Appellants' invention would somehow "prevent differential power analysis" is simply a misnomer.

Appeal Brief Section E.

In section 17 beginning on page 14 of the Examiner's Answer, the Examiner attempts to deflect the accusation that he is interpreting the term "instruction" in the claims to be equivalent to an entire routine by arguing that not only is his interpretation correct, but the interpretation is not necessary. The Examiner essentially argues that "an instruction" may be indeed equated with several lines of high-level program code (steps 5-8 in Figure 8 and the five lines of code that make up the MonPro routine – see Figure 10).

On page 15, line 20 of the Examiner's Answer, the Examiner summarizes the relevant code from Figure 8. However, even in the Examiner's compacted summary, the MonPro routine itself still constitutes a further five lines of high-level code. This is not an example of what those of ordinary skill in the art would understand an "instruction" to

be. How or why the Examiner believes this to be a disclosure of an “instruction” is simply not seen.

In a fallback position, the Examiner then asserts that his interpretation is not necessary, since “the conditional-write data processing instruction would be the machine instruction which ultimately writes the result of MonPro (M,x) subroutine to the appropriate register.” Here, the Examiner is not only changing his definition of what is supposed to be equivalent to the feature of the present invention, but this interpretation immediately breaks down because the “machine instruction” would not “[encode] condition codes specifying conditions . . .” as required by Appellants’ independent claims.

Moreover, in the paragraph bridging pages 16 and 17, the Examiner attempts to argue that his interpretation is consistent with the description in the present application. The Examiner argues that “[j]ust as a BEQ (branch upon equal) instruction encodes the behavior that the instruction will be performed if a flag is set and not performed if the flag is not set, so too does Qiu’s multiplication instruction [depending on a bit of the private key].” In actuality, the cited portion of Appellants’ specification at page 8, lines 14-17, states that “[t]his instruction encodes the behaviour that the specific branch will be performed if the flag indicating an equal result from previous processing is set and will be suppressed if this flag is not set.”

As those of ordinary skill in the art will understand, the flag referred to is a “condition code” within the processor core which is certainly not the same as a particular bit of a private key stored in memory as in the Qiu reference.

SUMMARY

Thus, the rebuttal of the Examiner's assertions that all claimed elements and all claimed interrelationships between elements set out in Appellants' independent claims are anticipated by the Qiu reference is not traversed by the Examiner's attempts at confusion in the outstanding Examiner's Answer. There is no need to include the terms SPA and DPA in Appellants' claims as long as the combination of elements set out in Appellants' claims does defeat differential power analysis. No other combination has been identified in any prior art which defeats DPA other than the claimed invention. The Examiner has not established how the Qiu reference, which admittedly can defeat SPA, but there is no indication that it anticipates or renders obvious any combination of elements (let alone the elements recited in Appellants' independent claims 1 and 6) that will defeat differential power analysis which is the problem discussed in the Background of the Invention of the present specification.


As a result of the above, there is simply no support for the rejection of Appellants' independent claims 1 and 6 or claims dependent thereon under 35 USC §102 or §103. Thus, and in view of the above, the rejection of claims 1-10 is clearly in error and reversal thereof by this Honorable Board is respectfully requested.

EVRARD et al.
Serial No. 10/527,812

Respectfully submitted,

NIXON & VANDERHYE P.C.

By:



Stanley C. Spooner
Reg. No. 27,393

SCS:kmm
901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100

Attachment:

http://en.wikipedia.org/wiki/Condition_Code_Register

Instruction (computer science)

Make a donation to Wikipedia and give the gift of knowledge!

From Wikipedia, the free encyclopedia

In computer science, an **instruction** is a single operation of a processor defined by an instruction set architecture. In a broader sense, an "instruction" may be any representation of an element of an executable program, such as a bytecode.

On traditional architectures, an instruction includes an opcode specifying the operation to be performed, such as "add contents of memory to register", and zero or more operand specifiers, which may specify registers, memory locations, or literal data. The operand specifiers may have addressing modes determining their meaning or may be in fixed fields.

In very long instruction word (VLIW) architectures, which include many microcode architectures, multiple simultaneous operations and operands are specified in a single instruction.

The size or length of an instruction varies widely, from as little as four bits in some microcontrollers to many hundreds of bits in some VLIW systems. Most modern processors used in personal computers, mainframes, and supercomputers have instruction sizes between 16 and 64 bits. In some architectures, notably most Reduced Instruction Set Computers, instructions are a fixed length, typically corresponding with that architecture's word size. In other architectures, instructions have variable length, typically integral multiples of a byte or a halfword.

The instructions constituting a program are rarely specified using their internal, numeric form; they may be specified by programmers using an assembly language or, more commonly, may be generated by compilers.

See also

- Command (computing)
- Data (computing)
- Machine language

Retrieved from "http://en.wikipedia.org/wiki/Instruction_%28computer_science%29"

Categories: Machine code

-
- This page was last modified on 3 May 2008, at 15:25.
 - All text is available under the terms of the GNU Free Documentation License. (See **Copyrights** for details.) Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a U.S. registered 501(c)(3) tax-deductible